VIT®

Vellore Institute of Technology

(Deemed to be University under section 3 of UGC Act, 1956)

Fall Semester 2025-26

Lab Assignment – 1

Slot: L13+L14

Class: VL2025260105679

Branch: B.tech CSBS

Course code & title: CBS3005

Cloud, Microservices and Applications

Faculty name: Nithya K

DA by:

Kartikey Gupta

22BBS0105

(i) Create and manage EC2 instances on AWS. Launch an application on an EC2 instance in one region and then migrate that instance to another region. Additionally, set up another EC2 instance to install MySQL, create a database (e.g., student or employee database), and perform basic SQL operations. (**5.0 marks**)

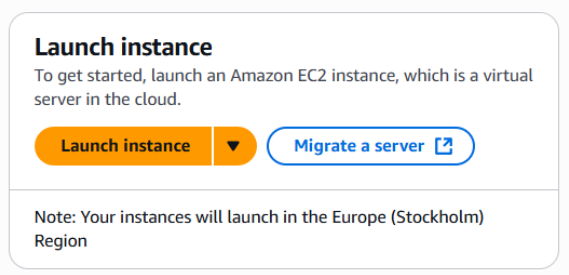
Part 1: Launch EC2 Instance for Static HTML App

1. Log in to AWS Console

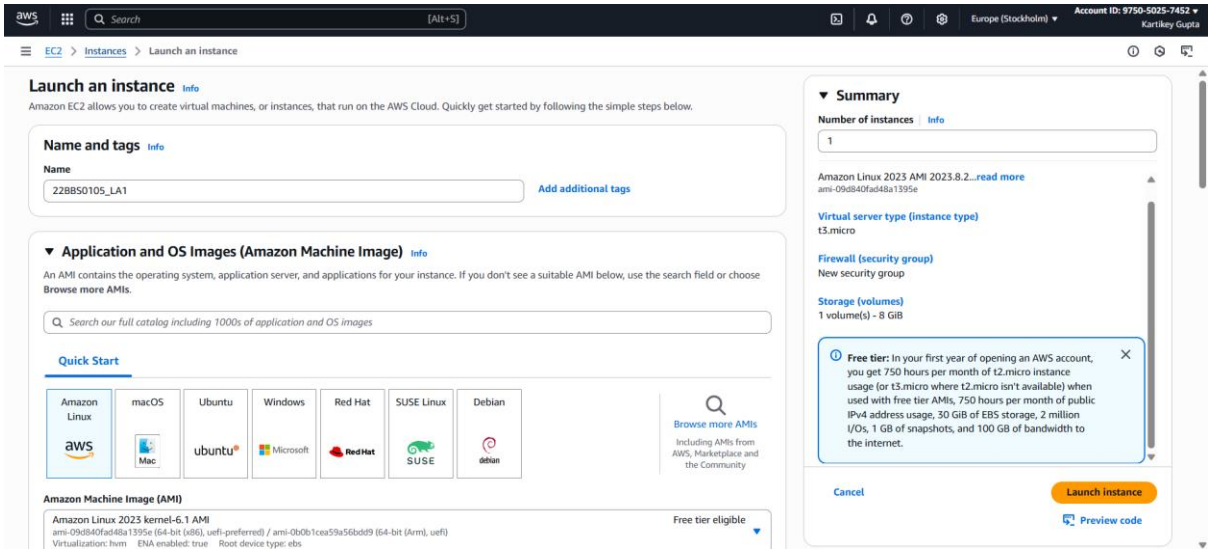
- Go to EC2 Dashboard

2. Launch EC2 Instance

- Click **Launch instance**

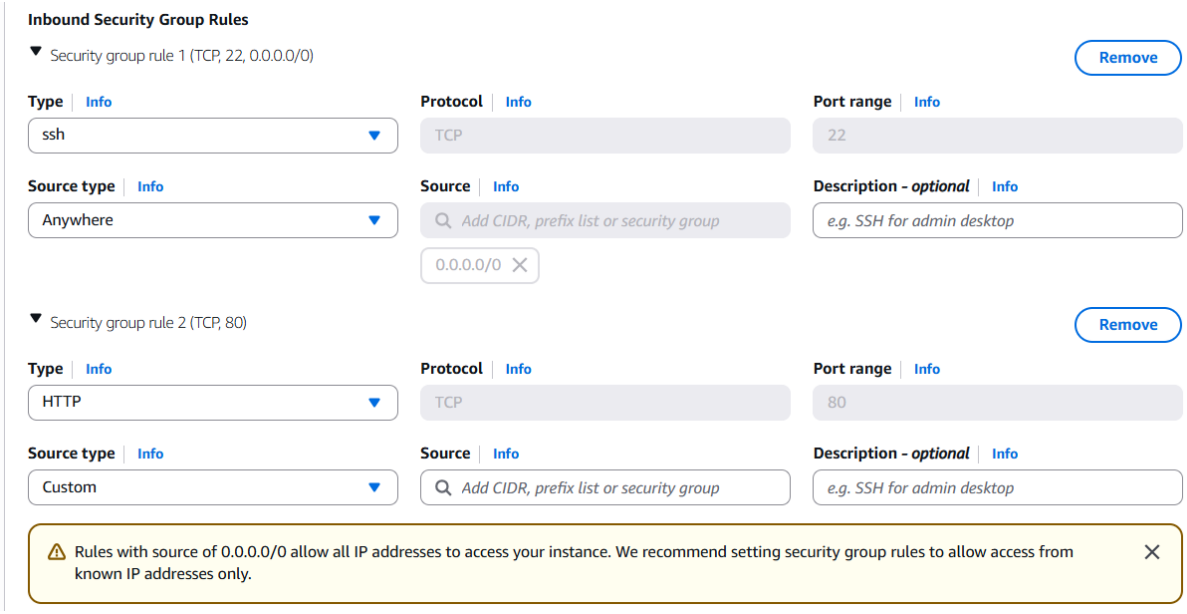


- **Name**: 22BBS0105_LA1
- **AMI**: Amazon Linux 2023 kernel-6.1
- **Instance type**: t3.micro

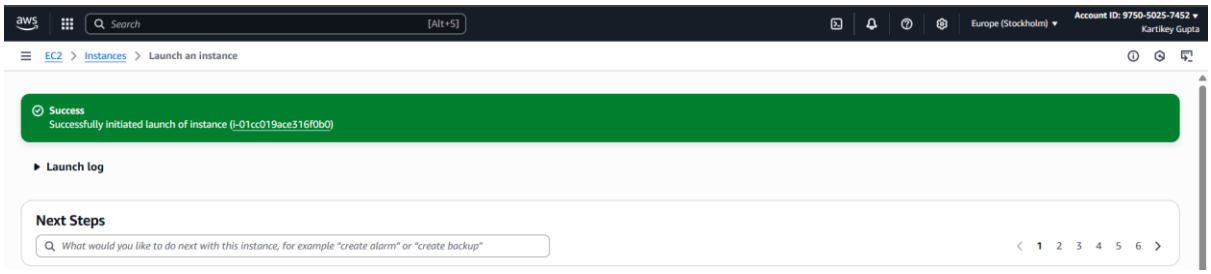


- **Key pair**: Create or select existing key
- **Network settings**:

- Click **Edit**
- Allow **SSH (port 22)** and **HTTP (port 80)**



- Click **Launch**



Step 2: Connecting to ec2 instance

Step 3: Installing Apache

```
[ec2-user@ip-172-31-28-55 ~]$ sudo dnf update -y
sudo dnf install httpd -y
sudo systemctl start httpd
sudo systemctl enable httpd
Amazon Linux 2023 Kernel Livepatch repository   173 kB/s |  19 kB     00:00

Dependencies resolved.
Nothing to do.
Complete!
Last metadata expiration check: 0:00:01 ago on Wed Aug  6 14:58:06 2025.
Dependencies resolved.
================================================================================
 Package              Arch     Version                    Repository     Size
================================================================================
Installing:
 httpd                x86_64   2.4.64-1.amzn2023.0.1      amazonlinux    47 k
Installing dependencies:
 apr                  x86_64   1.7.5-1.amzn2023.0.4       amazonlinux   129 k
 apr-util             x86_64   1.6.3-1.amzn2023.0.1       amazonlinux    98 k
 generic-logos-httpd  noarch   18.0.0-12.amzn2023.0.3     amazonlinux    19 k
 httpd-core           x86_64   2.4.64-1.amzn2023.0.1      amazonlinux   1.4 M
 httpd-filesystem     noarch   2.4.64-1.amzn2023.0.1      amazonlinux    13 k
 httpd-tools          x86_64   2.4.64-1.amzn2023.0.1      amazonlinux    81 k
 libbrotli            x86_64   1.0.9-4.amzn2023.0.2       amazonlinux   315 k
```

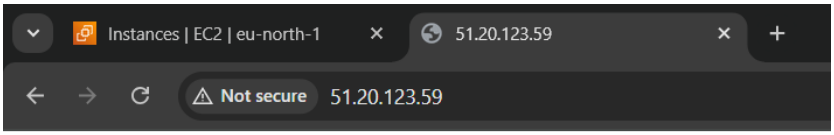
```
Install  12 Packages

Total download size: 2.3 M
Installed size: 6.9 M
Downloading Packages:
(1/12): apr-1.7.5-1.amzn2023.0.4.x86_64.rpm     3.5 MB/s | 129 kB     00:00
(2/12): apr-util-openssl-1.6.3-1.amzn2023.0.1.x 435 kB/s |  17 kB     00:00
(3/12): apr-util-1.6.3-1.amzn2023.0.1.x86_64.rp 2.3 MB/s |  98 kB     00:00
(4/12): generic-logos-httpd-18.0.0-12.amzn2023. 935 kB/s |  19 kB     00:00
(5/12): httpd-2.4.64-1.amzn2023.0.1.x86_64.rpm  2.0 MB/s |  47 kB     00:00
(6/12): httpd-core-2.4.64-1.amzn2023.0.1.x86_64  43 MB/s | 1.4 MB     00:00
(7/12): httpd-filesystem-2.4.64-1.amzn2023.0.1. 603 kB/s |  13 kB     00:00
(8/12): httpd-tools-2.4.64-1.amzn2023.0.1.x86_6 3.0 MB/s |  81 kB     00:00
(9/12): libbrotli-1.0.9-4.amzn2023.0.2.x86_64.r  12 MB/s | 315 kB     00:00
(10/12): mailcap-2.1.49-3.amzn2023.0.3.noarch.r 1.3 MB/s |  33 kB     00:00
(11/12): mod_http2-2.0.27-1.amzn2023.0.3.x86_64 6.2 MB/s | 166 kB     00:00
(12/12): mod_lua-2.4.64-1.amzn2023.0.1.x86_64.r 3.0 MB/s |  60 kB     00:00
--------------------------------------------------------------------------------
Total                                            16 MB/s | 2.3 MB     00:00
Running transaction check
Transaction check succeeded.
Running transaction test
Transaction test succeeded.
```

```
Installed:
  apr-1.7.5-1.amzn2023.0.4.x86_64
  apr-util-1.6.3-1.amzn2023.0.1.x86_64
  apr-util-openssl-1.6.3-1.amzn2023.0.1.x86_64
  generic-logos-httpd-18.0.0-12.amzn2023.0.3.noarch
  httpd-2.4.64-1.amzn2023.0.1.x86_64
  httpd-core-2.4.64-1.amzn2023.0.1.x86_64
  httpd-filesystem-2.4.64-1.amzn2023.0.1.noarch
  httpd-tools-2.4.64-1.amzn2023.0.1.x86_64
  libbrotli-1.0.9-4.amzn2023.0.2.x86_64
  mailcap-2.1.49-3.amzn2023.0.3.noarch
  mod_http2-2.0.27-1.amzn2023.0.3.x86_64
  mod_lua-2.4.64-1.amzn2023.0.1.x86_64

Complete!
Created symlink /etc/systemd/system/multi-user.target.wants/httpd.service → /usr
/lib/systemd/system/httpd.service.
[ec2-user@ip-172-31-28-55 ~]$
```

Checking if server is up and running



It works!

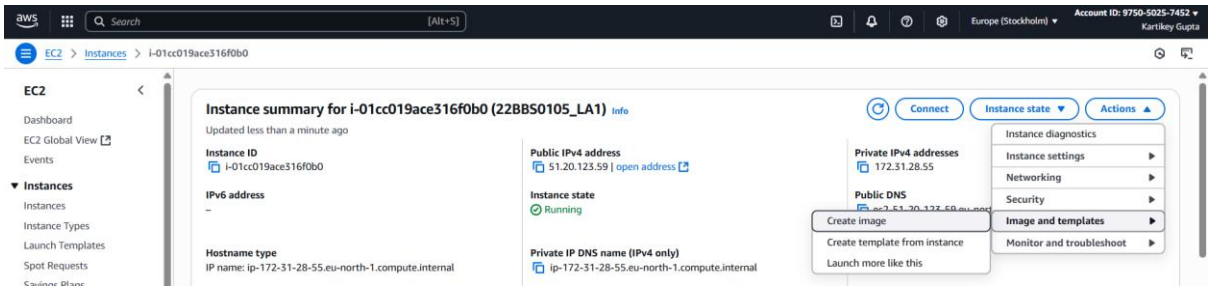
Working!!

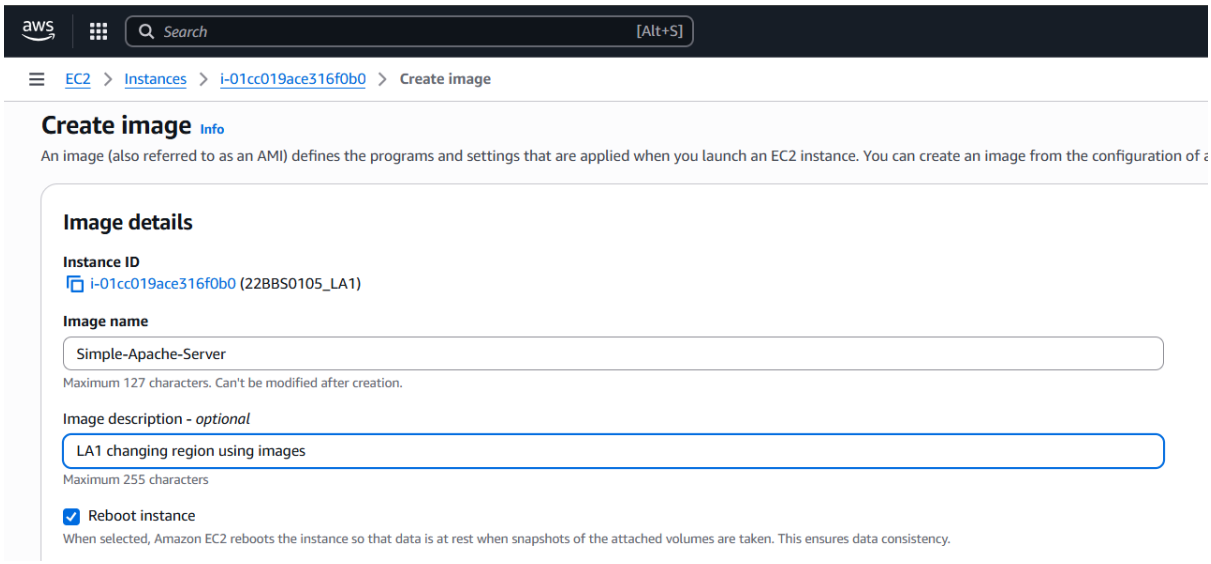
Part 2: Migrate the Instance to Another Region

AWS doesn't allow direct region change, so we will copy the AMI.

1. Create Image (AMI)

- Go to EC2 → Instances → Select instance → Actions → Image → Create Image

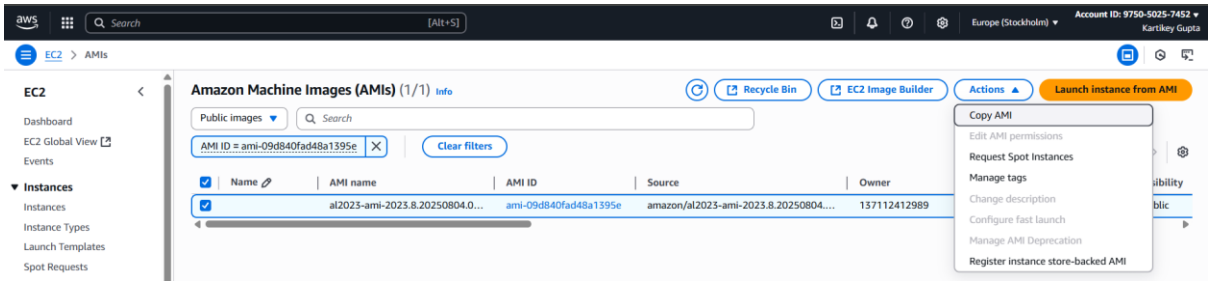




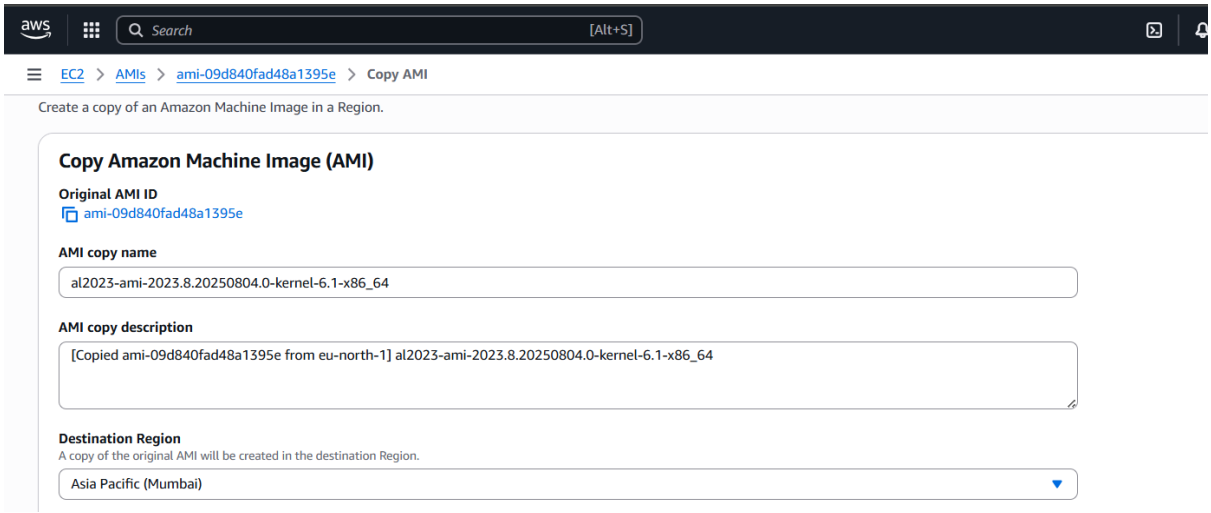
- Give a name and click **Create**

2. Copy AMI to Another Region

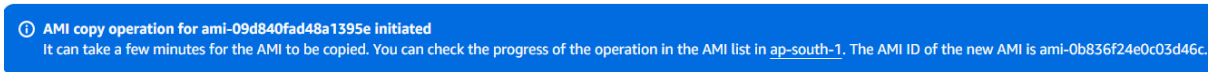
- Go to EC2 → AMIs
- Select your AMI → Actions → Copy AMI



- Choose **destination region** (Asia Pacific Mumbai)

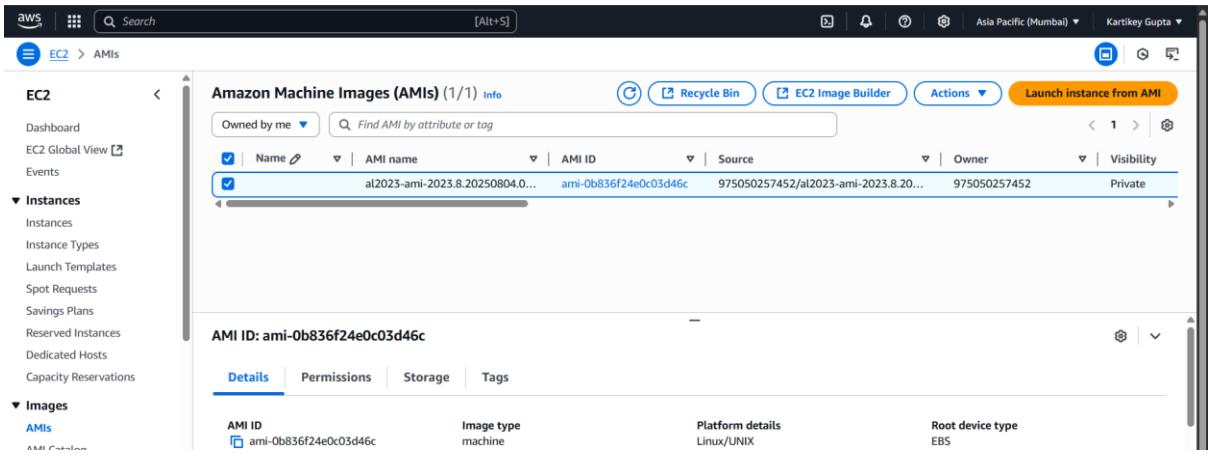


- Click **Copy**

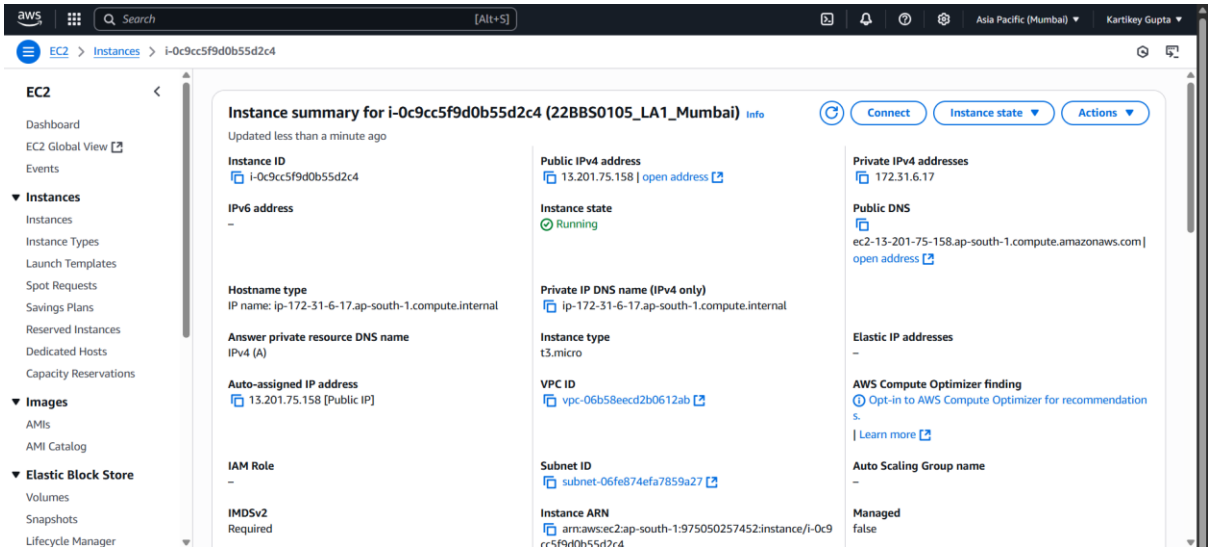


3. Launch New EC2 in Another Region

- Switch region to Mumbai
- Go to AMIs → Select copied AMI → Launch instance

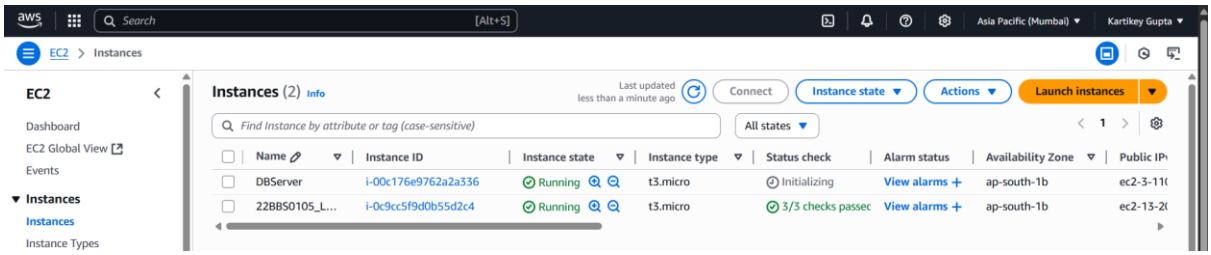


- Now app runs in the new region.



Part 3: Setup Database

Launch new instance with name DBServer following same steps above



Step 1: Connecting to DBServer

Step 2: Installing MySql

- sudo dnf update -y
- sudo dnf install mariadb105-server -y

```
[ec2-user@ip-172-31-8-160 ~]$ sudo dnf update -y
Amazon Linux 2023 Kernel Livepatch repository   187 kB/s |  19 kB     00:00
Dependencies resolved.
Nothing to do.
Complete!
[ec2-user@ip-172-31-8-160 ~]$ sudo dnf install mariadb105-server -y
Last metadata expiration check: 0:00:14 ago on Wed Aug  6 15:26:53 2025.
Dependencies resolved.
================================================================================
 Package                  Arch   Version                   Repository   Size
================================================================================
Installing:
 mariadb105-server        x86_64 3:10.5.29-1.amzn2023.0.1  amazonlinux  10 M
Installing dependencies:
 mariadb-connector-c      x86_64 3.3.10-1.amzn2023.0.1     amazonlinux 211 k
 mariadb-connector-c-config noarch 3.3.10-1.amzn2023.0.1   amazonlinux 9.9 k
 mariadb105               x86_64 3:10.5.29-1.amzn2023.0.1  amazonlinux 1.5 M
 mariadb105-common        x86_64 3:10.5.29-1.amzn2023.0.1  amazonlinux  28 k
 mariadb105-errmsg        x86_64 3:10.5.29-1.amzn2023.0.1  amazonlinux 212 k
 mysql-selinux            noarch 1.0.4-2.amzn2023.0.3      amazonlinux  36 k
```

Start and enable MySQL:

- sudo systemctl start mariadb
- sudo systemctl enable mariadb

Step 3: Log in to MySQL CLI

sudo mysql -u root -p

```
[ec2-user@ip-172-31-8-160 ~]$ sudo mysql -u root -p
Enter password:
Welcome to the MariaDB monitor.  Commands end with ; or \g.
Your MariaDB connection id is 3
Server version: 10.5.29-MariaDB MariaDB Server

Copyright (c) 2000, 2018, Oracle, MariaDB Corporation Ab and others.

Type 'help;' or '\h' for help. Type '\c' to clear the current input statement.

MariaDB [(none)]>
```

Step 4: Create Database and Table

```
CREATE DATABASE student;

USE student;

CREATE TABLE students (
  id INT AUTO_INCREMENT PRIMARY KEY,
  name VARCHAR(50),
  age INT
);
```

```
ec2-user@ip-172-31-8-160:~
MariaDB [(none)]> CREATE DATABASE student;
Query OK, 1 row affected (0.000 sec)

MariaDB [(none)]> USE student;
Database changed
MariaDB [student]> CREATE TABLE students (
    ->   id INT AUTO_INCREMENT PRIMARY KEY,
    ->   name VARCHAR(50),
    ->   age INT
    -> );
Query OK, 0 rows affected (0.008 sec)

MariaDB [student]> describe students
    -> ;
+-------+-------------+------+-----+---------+----------------+
| Field | Type        | Null | Key | Default | Extra          |
+-------+-------------+------+-----+---------+----------------+
| id    | int(11)     | NO   | PRI | NULL    | auto_increment |
| name  | varchar(50) | YES  |     | NULL    |                |
| age   | int(11)     | YES  |     | NULL    |                |
+-------+-------------+------+-----+---------+----------------+
3 rows in set (0.001 sec)
```

Step 5: Insert in table and display

INSERT INTO students (name, age) VALUES ('Aman', 20), ('Priya', 21), ('Ravi', 22);

```
MariaDB [student]> INSERT INTO students (name, age) VALUES ('Aman', 20), ('Priya
', 21), ('Ravi', 22);
Query OK, 3 rows affected (0.002 sec)
Records: 3  Duplicates: 0  Warnings: 0
```

SELECT * FROM students;

```
MariaDB [student]> select * from students
    -> ;
+----+-------+------+
| id | name  | age  |
+----+-------+------+
|  1 | Aman  |   20 |
|  2 | Priya |   21 |
|  3 | Ravi  |   22 |
+----+-------+------+
3 rows in set (0.000 sec)
```

Closing connections

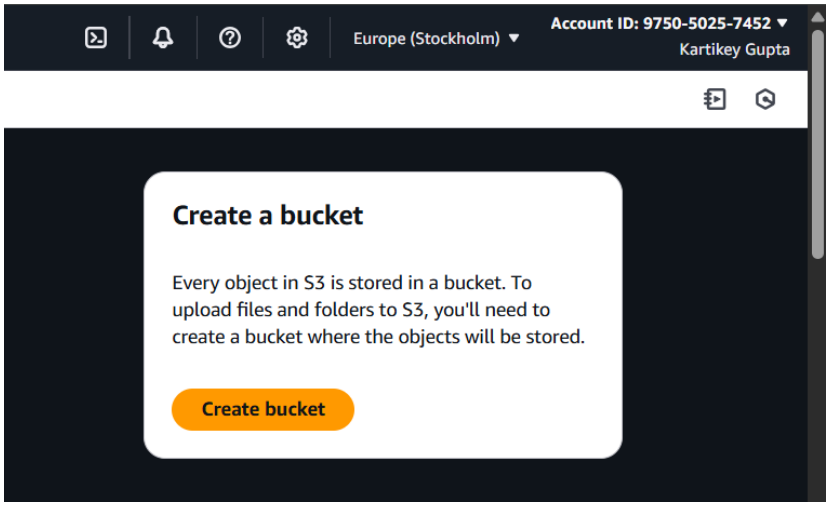
```
MariaDB [student]> exit
Bye
[ec2-user@ip-172-31-8-160 ~]$ exit
logout
Connection to 3.110.44.209 closed.

karti@Kartikey MINGW64 ~/OneDrive/desktop/sem 7/Cloud/Keys (main)
$
```

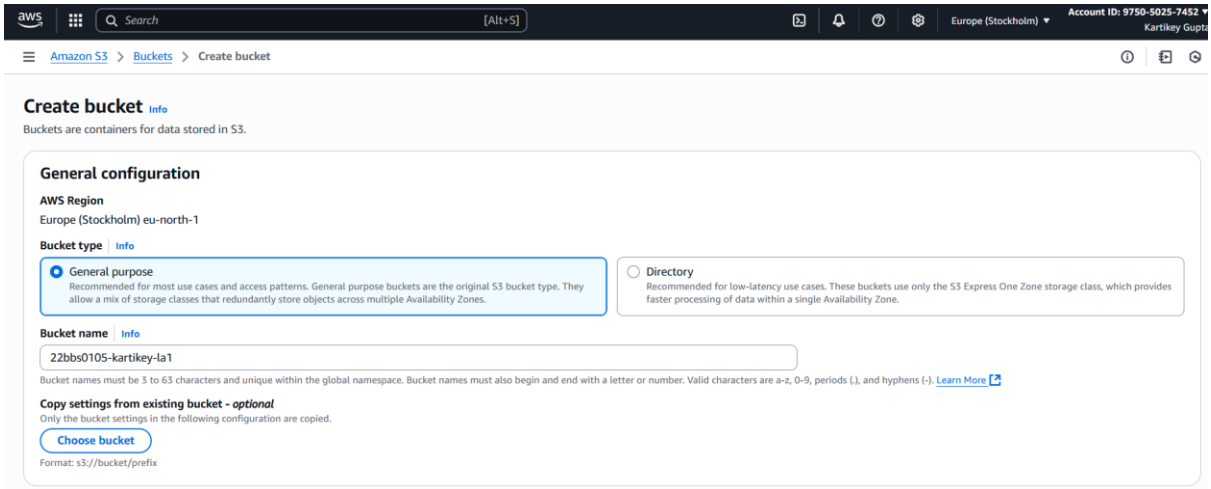
(ii) Set up an S3 bucket in AWS, ensuring the name is globally unique. Upload your static website files (HTML, CSS, JavaScript, etc.) to this bucket. Enable static website hosting in the S3 bucket properties and configure the documents. Apply the necessary bucket policy to make your site publicly accessible. Also, enable versioning on the bucket to maintain previous versions of your website files. (**5.0 marks**)

Step 1: Create an S3 Bucket

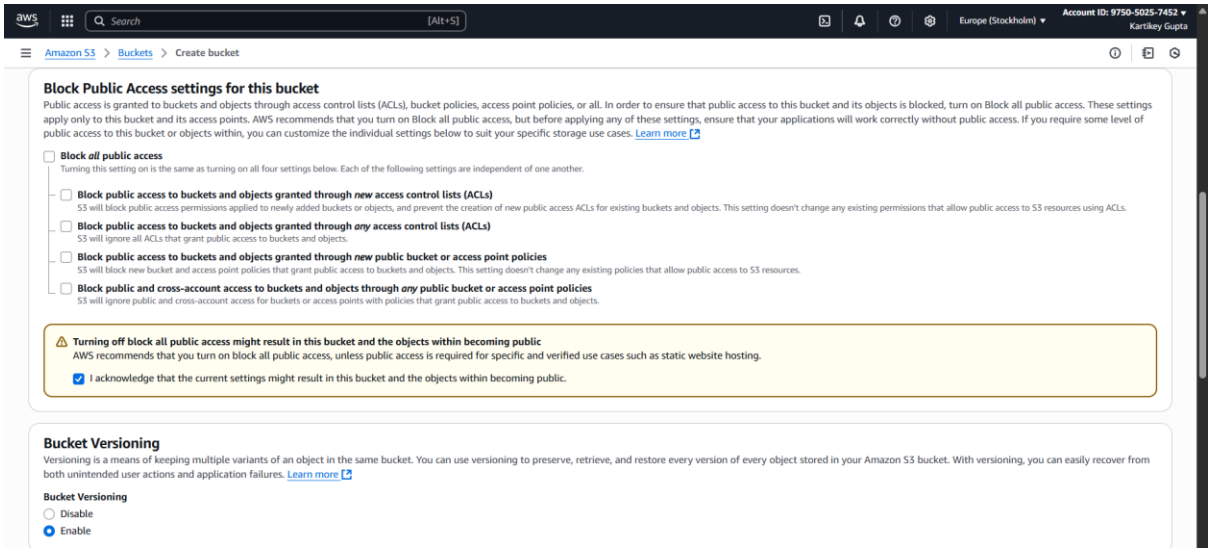
1. Go to S3 Console: https://s3.console.aws.amazon.com/s3/home
2. Click “Create bucket”



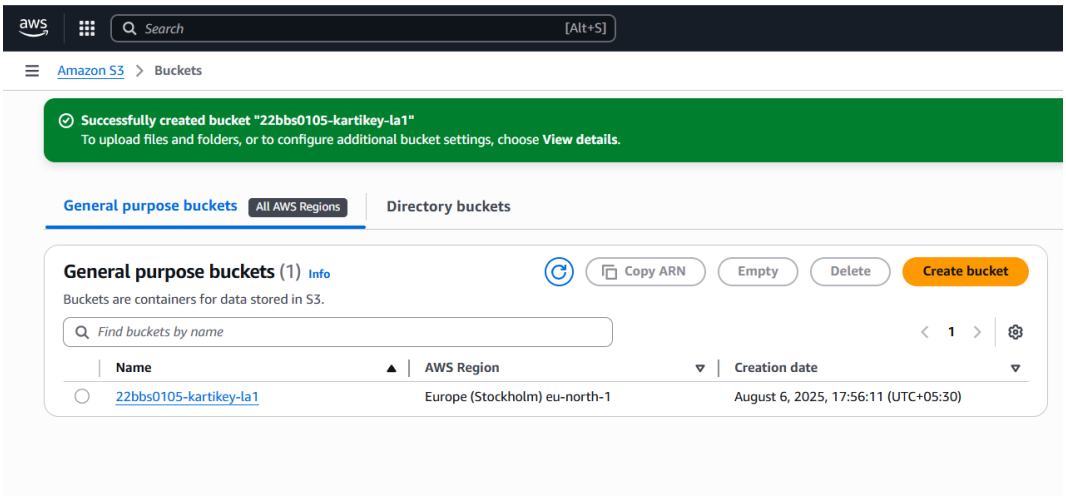
3. Enter a globally unique bucket name, 22bbs0105-kartikey-la1



4. Uncheck “Block all public access” and “Enable bucket versioning”

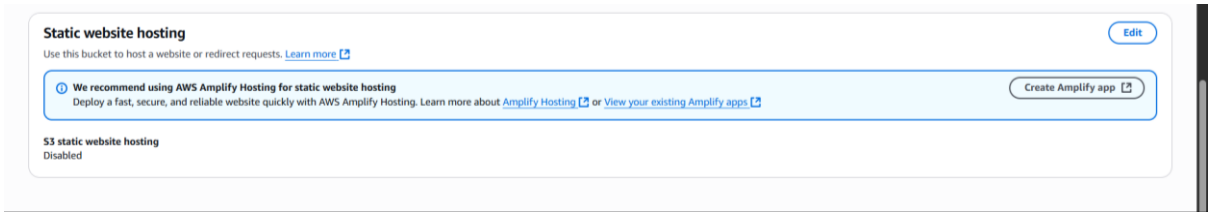


5. Click “Create bucket”



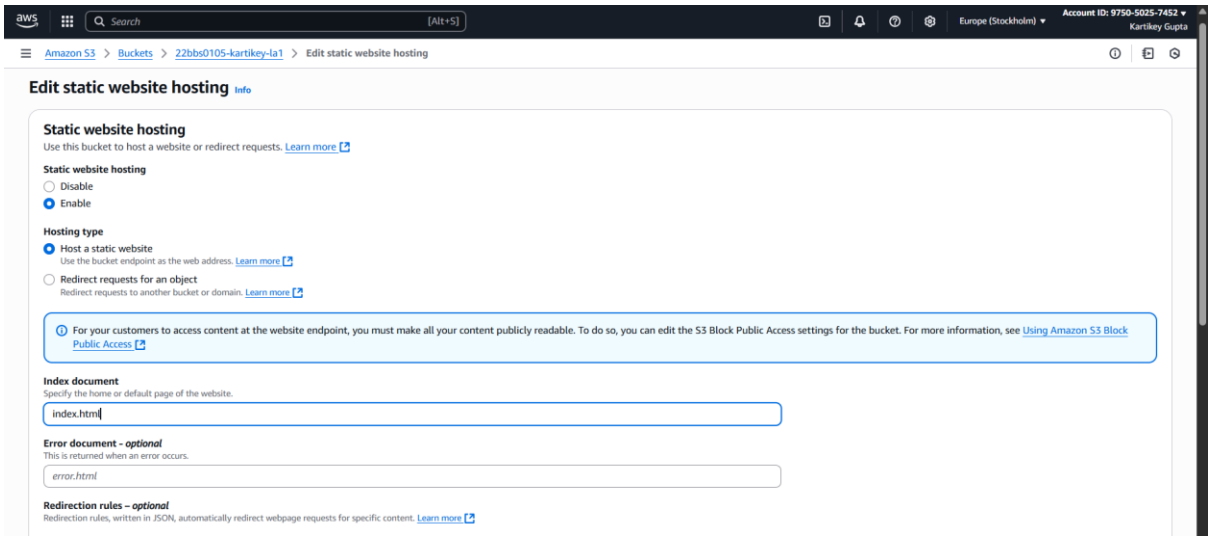
Step 2: Enable Static Website Hosting

1. Go to the created bucket
2. Click “Properties”



3. Scroll to “Static website hosting” and enable it
4. Select “Host a static website”
5. Set:

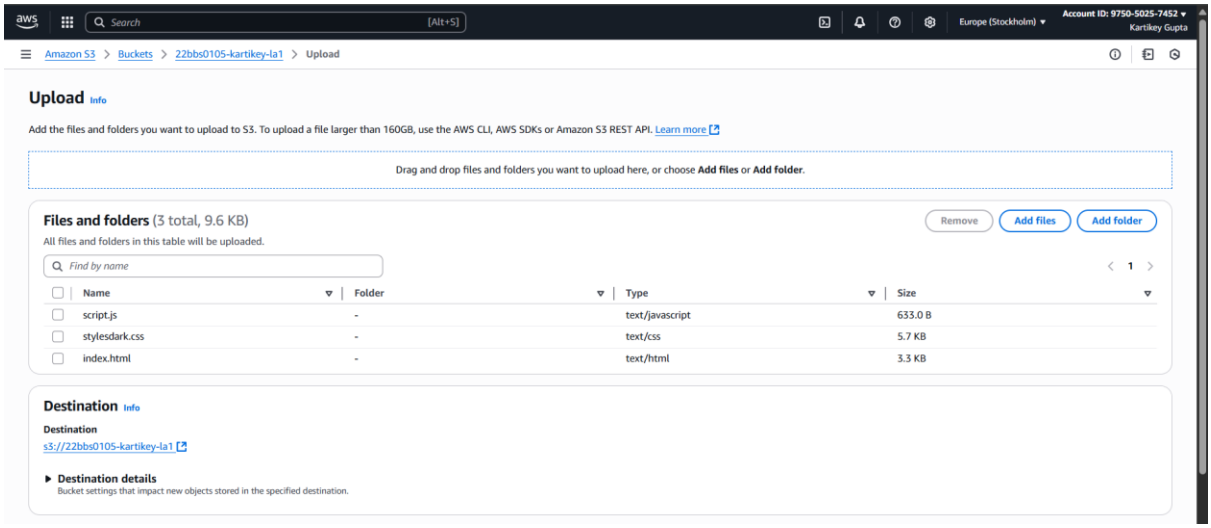
- Index document: index.html



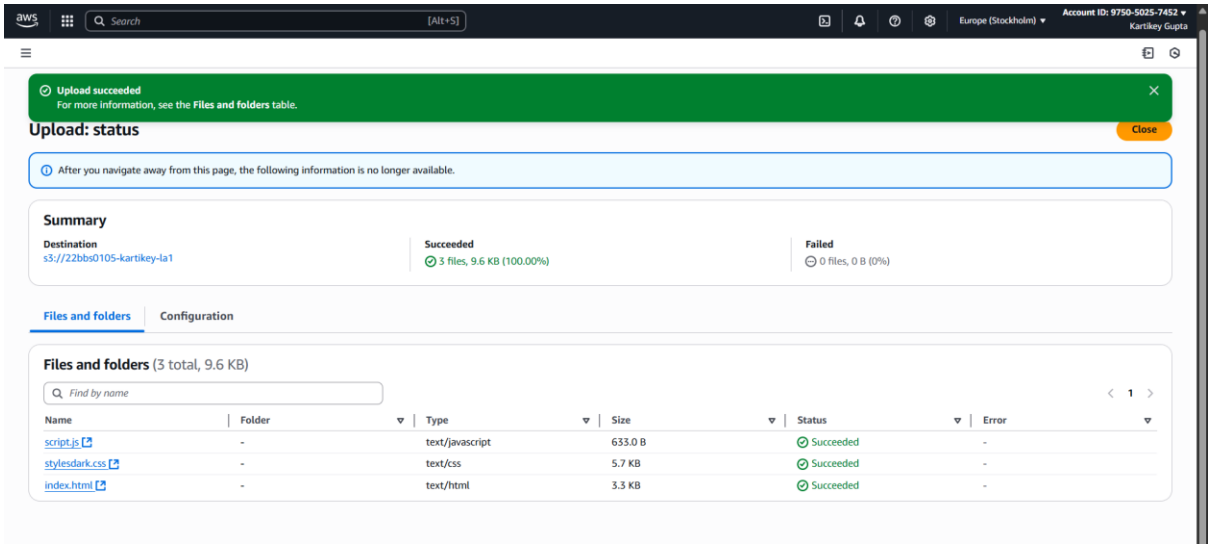
6. Save changes

Step 3: Upload Website Files

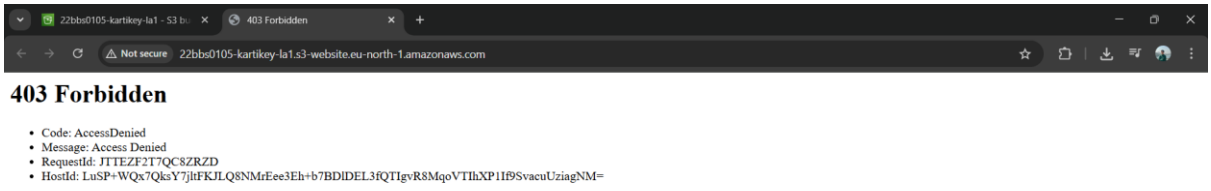
1. Go to the “Objects” tab in your bucket
2. Click “Upload”
3. Add index.html, style.css, script.js.



4. Click “Upload”

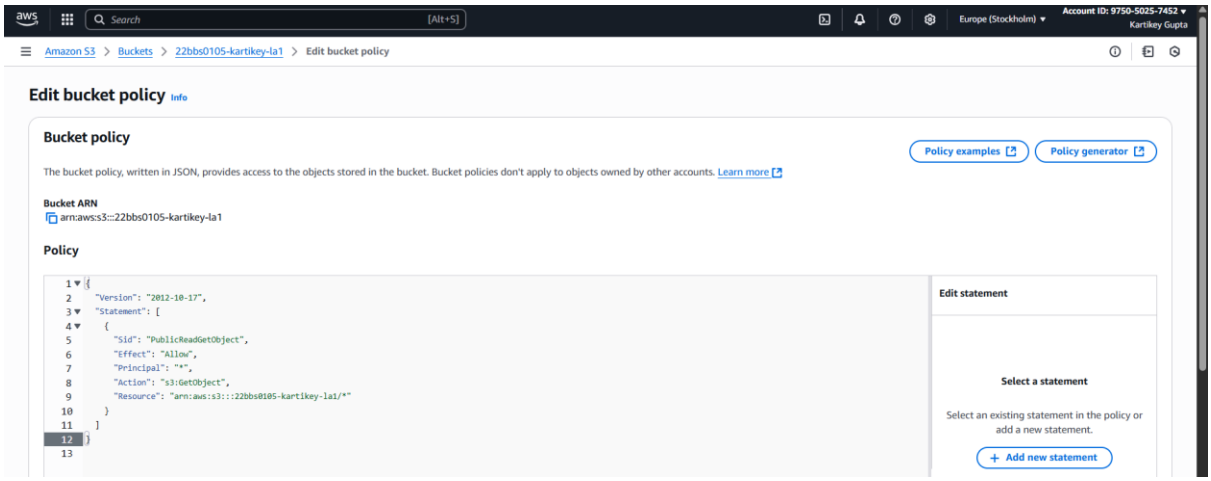


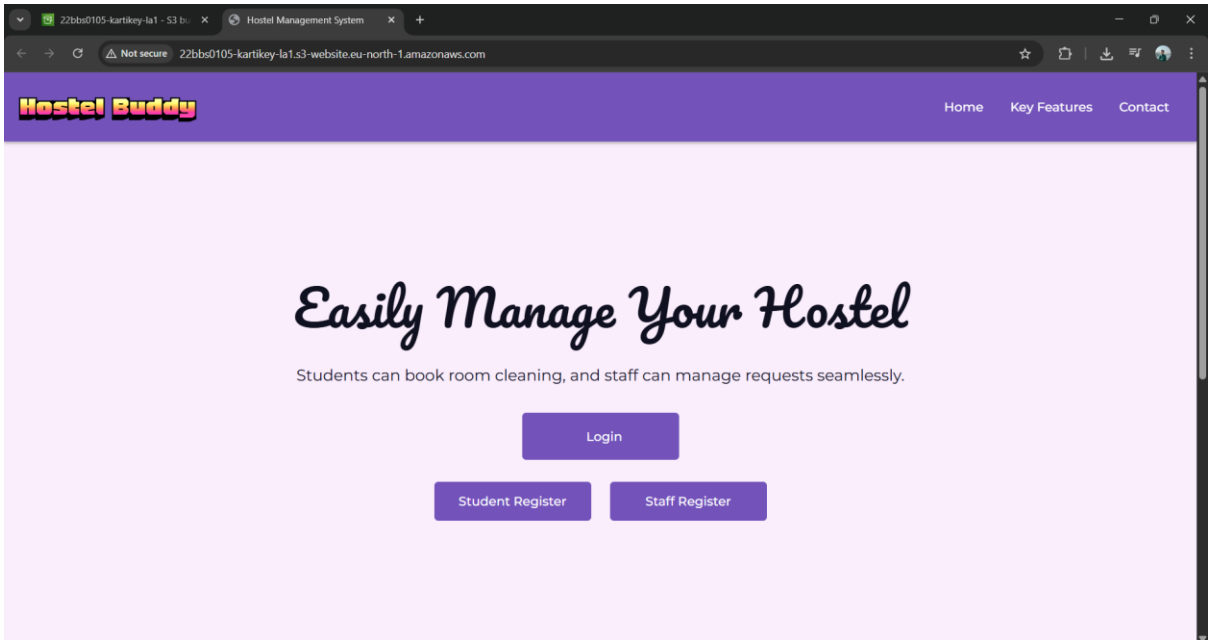
Permission Denied now



Step 4: Make Files Public (Bucket Policy)

1. Go to “Permissions” tab
2. Scroll to Bucket Policy





Website hosted successfully

Link: http://22bbs0105-kartikey-la1.s3-website.eu-north-1.amazonaws.com/